

**CLAIMS:**

What is claimed is:

- filed 5/1/92*
1. A method of locking a system resource, comprising:  
attempting to obtain a lock on the system resource;  
associating a hand-off lock with the lock on the system  
resource if the attempt to obtain the lock is unsuccessful;  
and
  - 10 obtaining the hand-off lock on the system resource if  
the attempt to obtain the lock on the system resource is  
unsuccessful.
  - 15 2. The method of claim 1, wherein the lock is a simple  
lock.
  - 20 3. The method of claim 1, wherein the hand-off lock is a  
krlock.
  - 25 4. The method of claim 1, wherein the step of attempting  
to obtain a lock on the system resource is performed a  
predetermined number of times before associating a hand-off  
lock with the lock on the system resource.
  - 30 5. The method of claim 1, wherein the hand-off lock is  
obtained from a pool of hand-off locks.
  6. The method of claim 1, wherein associating a hand-off  
lock with the lock on the system resource includes storing  
an index of the hand-off lock in a lock word of the lock on  
the system resource.

Docket No. AUS920000868US1

7. The method of claim 1, wherein if the lock on the system resource is freed, the method further comprises:

obtaining the lock on the system resource;

releasing the hand-off lock; and

5 handing-off the hand-off lock to a next processor spinning on the hand-off lock.

8. The method of claim 1, wherein the method is implemented in a multiprocessor system having one or more

10 nodes, and wherein the hand-off lock includes a per-node word which contains a state of the hand-off lock on each node of the multiprocessor system and a per-processor spin field for each processor of the multiprocessor system.

15 9. The method of claim 8, wherein when the lock on the system resource is released, the per-node word and per-processor spin fields of the hand-off lock are updated to reflect a next processor obtaining the lock on the system resource.

20

10. The method of claim 1, wherein the method is implemented in one of a SMP, a NUMA, and a ccNUMA system.

11. A computer program product in a computer readable 25 medium for locking a system resource, comprising:

first instructions for attempting to obtain a lock on the system resource;

second instructions for associating a hand-off lock with the lock on the system resource if the attempt to 30 obtain the lock is unsuccessful; and

Docket No. AUS920000868US1

third instructions for obtaining the hand-off lock on the system resource if the attempt to obtain the lock on the system resource is unsuccessful.

5 12. The computer program product of claim 11, wherein the lock is a simple lock.

13. The computer program product of claim 11, wherein the hand-off lock is a krlock.

10

14. The computer program product of claim 11, wherein the first instructions are executed a predetermined number of times before the second instructions are executed.

15

15. The computer program product of claim 11, wherein the hand-off lock is obtained from a pool of hand-off locks.

16. The computer program product of claim 11, wherein the second instructions include instructions for storing an

20

index of the hand-off lock in a lock word of the lock on the system resource.

17. The computer program product of claim 11, wherein the computer program product further comprises:

25

fourth instructions for obtaining the lock on the system resource, if the lock on the system resource is freed;

fifth instructions for releasing the hand-off lock; and sixth instructions for handing-off the hand-off lock to

30

a next processor spinning on the hand-off lock.

Docket No. AUS920000868US1

18. The computer program product claim 11, wherein the computer program product is executed in a multiprocessor system having one or more nodes, and wherein the hand-off lock includes a per-node word which contains a state of the  
5 hand-off lock on each node of the multiprocessor system and a per-processor spin field for each processor of the multiprocessor system.

19. The computer program product of claim 18, further  
10 comprising fourth instructions for updating the per-node word and per-processor spin fields of the hand-off lock to reflect a next processor obtaining the lock on the system resource, when the lock on the system resource is released.

15 20. The computer program product of claim 11, wherein the first, second and third instructions are formatted for use with one of an SMP, a NUMA, and a ccNUMA system.

21. An apparatus for locking a system resource, comprising:  
20 means for attempting to obtain a lock on the system resource;

means for associating a hand-off lock with the lock on the system resource if the attempt to obtain the lock is unsuccessful; and

25 means for obtaining the hand-off lock on the system resource if the attempt to obtain the lock on the system resource is unsuccessful.

22. The apparatus of claim 21, wherein the lock is a simple  
30 lock.

Docket No. AUS920000868US1

23. The apparatus of claim 21, wherein the hand-off lock is a krlock.

24. The apparatus of claim 21, wherein the means for  
5 attempting to obtain the lock on the system resource  
operates a predetermined number of times before the means  
for associating the hand-off lock operates.

25. The apparatus of claim 21, wherein the hand-off lock is  
10 obtained from a pool of hand-off locks.

26. The apparatus of claim 21, wherein the means for  
associating the hand-off lock includes means for storing an  
index of the hand-off lock in a lock word of the lock on the  
15 system resource.

27. The apparatus of claim 21, wherein the apparatus  
further comprises:

means for obtaining the lock on the system resource, if  
20 the lock on the system resource is freed;  
means for releasing the hand-off lock; and  
means for handing-off the hand-off lock to a next  
processor spinning on the hand-off lock.

25 28. The apparatus claim 21, wherein the apparatus is part  
of a multiprocessor system having one or more nodes, and  
wherein the hand-off lock includes a per-node word which  
contains a state of the hand-off lock on each node of the  
multiprocessor system and a per-processor spin field for  
30 each processor of the multiprocessor system.

Docket No. AUS920000868US1

29. The apparatus of claim 28, further comprising means for updating the per-node word and per-processor spin fields of the hand-off lock to reflect a next processor obtaining the lock on the system resource, when the lock on the system  
5 resource is released.

30. The apparatus of claim 21, wherein the apparatus is part of one of a SMP, a NUMA, and a ccNUMA system.